



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER OF PATENTS AND TRADEMARKS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/374,374	08/13/1999	DANIEL M. COFFMAN	Y0999-276-(8	3661

7590 05/28/2003

FRANK CHAU ESQ  
F CHAU & ASSOCIATES LLP  
SUITE 501  
1900 HEMPSTEAD TURNPIKE  
EAST MEADOW, NY 11554

EXAMINER

ARMSTRONG, ANGELA A

ART UNIT	PAPER NUMBER
----------	--------------

2654

DATE MAILED: 05/28/2003

16

Please find below and/or attached an Office communication concerning this application or proceeding.

16

# Office Action Summary

Application No.

09/374,374

Applicant(s)

COFFMAN ET AL. (b)

Examiner

Angela A. Armstrong

Art Unit

2654

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

## Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133).
- Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

## Status

- 1) ☒ Responsive to communication(s) filed on 24 March 2003.
- 2a) ☒ This action is **FINAL**.                      2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

## Disposition of Claims

- 4) ☒ Claim(s) 1-25 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-25 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

## Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
- Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
- 11) ☐ The proposed drawing correction filed on \_\_\_\_\_ is: a) ☐ approved b) ☐ disapproved by the Examiner.
- If approved, corrected drawings are required in reply to this Office action.
- 12) ☐ The oath or declaration is objected to by the Examiner.

## Priority under 35 U.S.C. §§ 119 and 120

- 13) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.
- 14) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. § 119(e) (to a provisional application).
- a) ☐ The translation of the foreign language provisional application has been received.
- 15) ☐ Acknowledgment is made of a claim for domestic priority under 35 U.S.C. §§ 120 and/or 121.

## Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449) Paper No(s) \_\_\_\_\_.
- 4) ☐ Interview Summary (PTO-413) Paper No(s). \_\_\_\_\_.
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: \_\_\_\_\_.

**DETAILED ACTION**

***Claim Rejections - 35 USC § 103***

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

1. Claims 1-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Johnson (US Patent No. 5,748,974) in view of Eberman et al (US Patent No. 5,805,775) in further view of Haddock (US Patent No. 5,265,014).

Johnson discloses a multi-modal natural language interface for cross-application tasks which interprets user requests combining natural language input from the user with information selected from a current application and sends the request in the proper form to an appropriate auxiliary application for processing.

Regarding claim 1, at col. 3, line 63 continuing to col. 4, line 20, Johnson teaches the system sends the output of a speech recognizer and non-speech input received by a screen manager to a dispatcher. Based on the output of the natural language processor, the dispatcher invokes the application manager to determine which application should process the request, which reads on "presenting a command associated with an application to a dialog manager, the application associated with the command being unknown to the dialog manager."

Johnson fails to specifically disclose, "the application associated with the command being one of a plurality of active applications." However, implementation of a

Art Unit: 2654

computer system, which allows a user to interact with a plurality of applications using natural language, was well known in the art.

In a similar field of endeavor, Eberman teaches an application user interface which allows the user to control the activities of multiple applications which are executed by the processors under control of the operating software system (col. 3, lines 25-39).

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to allow for the presented command to be associated with one of a plurality of active applications, as suggested by Eberman, for the purpose of allowing the user the convenience of accessing the processing of all applications to have a request processed automatically, without the user having to manually access any of the applications.

Johnson fails to specifically disclose "the dialog manager determining current context of the command by reviewing a multi-modal history of events." However, determining current context of user input responses by reviewing a history of events was well known in the art.

In a similar field of endeavor, Haddock discloses a multi-modal user interface for removing a referential ambiguity from a natural language input to a computer system. Specifically, Haddock teaches a user providing both a textual and a referential input and implements a history space or list of successive queries (col. 4, lines 61-66). The system functions to remove any ambiguity from a user query by referring to the history of the dialogue between the user and the system (col. 5, line 33 continuing to col. 6, line 64) to generate an output query that can be translated into a formal database query such as an SQL query which is then directed to the database to obtain a response. Haddock teaches

Art Unit: 2654

the system is advantageous in providing a user-friendly way to resolve ambiguity in a natural language system (col. 2, lines 46-50).

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to implement determining current context of user input responses by reviewing a history of events as suggested by Haddock, for the purpose of providing a user-friendly way to resolve ambiguity in a natural language system, as also suggested by Haddock.

Additionally, at col. 4, lines 29-52, Johnson teaches the application manager determines which application is applicable to the user input, sends a request to a code generator for the application, and generates the application program code interface code required to access the information, which reads on “determining at least one method responsive to the command based on the current context.”

Additionally, at col. 4, lines 48-52, Johnson discloses a response generator that generates a response appropriate to the nature of the request submitted by the user and the current application, which reads on “executing the at least one method responsive to the command associated with the application.”

Additionally, at col. 4, line 17-53 and col. 5, line 23-29, Johnson teaches “changing dialog focus,” because the system ascertains that a different application should process the user request, and utilizes the different application’s resources to process the request, since the procedures to access a new or different application requires the system to access or focus on the new or different application. The system of Johnson performs this processing without requiring the user to perform the task of accessing the application,

Art Unit: 2654

but the application is indeed accessed and opened to allow for the processing of the user's request.

Regarding claim 2, Johnson, Eberman and Haddock teach everything as claimed in claim 1. Additionally, at col. 3, lines 33-42, Johnson teaches the multi-modal interface is linked to the applications and allows for combining input from various modalities, which reads on "employing at least one multi-modal device for presenting the command."

Regarding claim 3, Johnson, Eberman and Haddock teaches everything as claimed in claim 2. Additionally, at col. 3, lines 37-42, Johnson discloses the input means can be via speech, typed or handwritten and includes point-and-click, touch and keyboard inputs, which reads on "multi-modal device is a computer."

Regarding claim 4, Johnson, Eberman and Haddock teaches everything as claimed in claim 1. Johnson does not specifically teach providing a linked list of all events in the multi-modal history. However, providing a linked history of dialog between a user and a system was well known in the art.

Haddock discloses the history of the dialog between the user and the computer system as a history space or list of successive queries (col. 4, lines 61-66), which reads on "linked list of all events."

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to implement a linked list of events in the multi-modal history, as suggested by Haddock, for the purpose of removing the ambiguity of a users input query, as also suggested by Haddock.

Art Unit: 2654

Regarding claim 5, Johnson, Eberman and Haddock teach everything as claimed in claim 4. Johnson does not specifically teach events in a multi-modal history include events linked by time, type, transaction, class or dialog focus. However, providing linked history events of a dialog between a user and a natural language system was well known.

Haddock discloses the history of the dialog between the user and the computer system as a history space or list of successive queries (col. 4, lines 61-66), which reads on "events linked by time."

Regarding claim 6, Johnson, Eberman and Haddock teach everything as claimed in claim 1. Additionally, at col. 5, line 55 continuing to col. 6, line 19, Johnson teaches referencing a concept/application table in which for a given input the application manager finds all concepts in the representation and determines each application registered with every concept in the representation to create an application set. The semantic representation is sent to the applications code generator to allow the application to act on the code and retrieve the requested data, which reads on "referencing all active applications using a component control to determine the at least one method which is appropriate."

Regarding claim 7, Johnson, Eberman and Haddock teach everything as claimed in claim 1. Additionally, at col. 6, lines 20-40, Johnson teaches the semantic representation of the users input is sent to the applications code generator to allow the application to act on the code and retrieve the requested data, which reads on "the command is presented in a formal language such that a plurality of human utterances represent an action to be taken."

Art Unit: 2654

Regarding claim 8, Johnson, Eberman and Haddock teaches everything as claimed in claim 1. Additionally, at col. 3, lines 60-62, Johnson teaches maintaining item in focus information, which reads on “maintaining a current dialog focus.” At col. 5, line 61 continuing to col. 6, line 9, Johnson teaches concept/application table which reads on “a list of expected responses to provide a reference for determining the current context.”

Regarding claim 9, Johnson, Eberman and Haddock teach everything as claimed in claim 1. Additionally, at col. 4, lines 9-17 and col. 5, lines 9-11, Johnson teaches the natural language processor interacts with the user to clarify missing information or ambiguities, which reads on “querying a user for one of information needed to resolve the current context and information needed to take an appropriate action.

Regarding claim 10, at col. 2, line 62 continuing to col. 3, line 16, Johnson teaches the hardware environment used in the multi-modal system, which includes running an operating system supporting multi-tasking and includes a central processing unit and Ram for storing application program code and data, which reads on “program storage device readable by machine, tangibly embodying a program of instructions executable by the machine.”

Additionally, at col. 3, line 63 continuing to col. 4, line 20, Johnson teaches the system sends the output of a speech recognizer and non-speech input received by a screen manager to a dispatcher. Based on the output of the natural language processor, the dispatcher invokes the application manager to determine which application should process the request, which reads on “presenting a command associated with an application to a dialog manager, the application associated with the command being unknown to the dialog manager.”



Art Unit: 2654

Johnson fails to specifically disclose, “the application associated with the command being one of a plurality of active applications.” However, implementation of a computer system, which allows a user to interact with a plurality of applications using natural language, was well known in the art.

In a similar field of endeavor, Eberman teaches an application user interface which allows the user to control the activities of multiple applications which are executed by the processors under control of the operating software system (col. 3, lines 25-39).

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to allow for the presented command to be associated with one of a plurality of active applications, as suggested by Eberman, for the purpose of allowing the user the convenience of accessing the processing of all applications to have a request processed automatically, without the user having to manually access any of the applications.

Johnson fails to specifically disclose “the dialog manager determining current context of the command by reviewing a multi-modal history of events.” However, determining current context of user input responses by reviewing a history of events was well known in the art.

In a similar field of endeavor, Haddock discloses a multi-modal user interface for removing a referential ambiguity from a natural language input to a computer system. Specifically, Haddock teaches a user providing both a textual and a referential input and implements a history space or list of successive queries (col. 4, lines 61-66). The system functions to remove any ambiguity from a user query by referring to the history of the dialogue between the user and the system (col. 5, line 33 continuing to col. 6, line 64) to

Art Unit: 2654

generate an output query that can be translated into a formal database query such as an SQL query which is then directed to the database to obtain a response. Haddock teaches the system is advantageous in providing a user-friendly way to resolve ambiguity in a natural language system (col. 2, lines 46-50).

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to implement determining current context of user input responses by reviewing a history of events as suggested by Haddock, for the purpose of providing a user-friendly way to resolve ambiguity in a natural language system, as also suggested by Haddock.

Additionally, at col. 4, lines 29-52, Johnson teaches the application manager determines which application is applicable to the user input, sends a request to a code generator for the application, and generates the application program code interface code required to access the information, which reads on “determining at least one method responsive to the command based on the current context.”

Additionally, at col. 4, lines 48-52, Johnson discloses a response generator that generates a response appropriate to the nature of the request submitted by the user and the current application, which reads on “executing the at least one method responsive to the command associated with the application.”

Additionally, at col. 4, line 17-53 and col. 5, line 23-29, Johnson teaches “changing dialog focus,” because the system ascertains that a different application should process the user request, and utilizes the different application’s resources to process the request, since the procedures to access a new or different application requires the system to access or focus on the new or different application. The system of Johnson performs

Art Unit: 2654

this processing without requiring the user to perform the task of accessing the application, but the application is indeed accessed and opened to allow for the processing of the user's request.

Regarding claim 11, Johnson, Eberman and Haddock teach everything as claimed in claim 10. Additionally, at col. 3, lines 33-42, Johnson teaches the multi-modal interface is linked to the applications and allows for combining input from various modalities, which reads on "employing at least one multi-modal device for presenting the command."

Regarding claim 12, Johnson, Eberman and Haddock teaches everything as claimed in claim 11. Additionally, at col. 3, lines 37-42, Johnson discloses the input means can be via speech, typed or handwritten and includes point-and-click, touch and keyboard inputs, which reads on "multi-modal device is a computer."

Regarding claim 13, Johnson, Eberman and Haddock teaches everything as claimed in claim 10. Johnson does not specifically teach providing a linked list of all events in the multi-modal history. However, providing a linked history of dialog between a user and a system was well known in the art.

Haddock discloses the history of the dialog between the user and the computer system as a history space or list of successive queries (col. 4, lines 61-66), which reads on "linked list of all events."

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to implement a linked list of events in the multi-modal history, as suggested by Haddock, for the purpose of removing the ambiguity of a users input query, as also suggested by Haddock.

Art Unit: 2654

Regarding claim 14, Johnson, Eberman and Haddock teach everything as claimed in claim 13. Johnson does not specifically teach events in a multi-modal history include events linked by time, type, transaction, class or dialog focus. However, providing linked history events of a dialog between a user and a natural language system was well known.

Haddock discloses the history of the dialog between the user and the computer system as a history space or list of successive queries (col. 4, lines 61-66), which reads on "events linked by time."

Regarding claim 15, Johnson, Eberman and Haddock teach everything as claimed in claim 10. Additionally, at col. 5, line 55 continuing to col. 6, line 19, Johnson teaches referencing a concept/application table in which for a given input the application manager finds all concepts in the representation and determines each application registered with every concept in the representation to create an application set. The semantic representation is sent to the applications code generator to allow the application to act on the code and retrieve the requested data, which reads on "referencing all active applications using a component control to determine the at least one method which is appropriate."

Regarding claim 16, Johnson, Eberman and Haddock teach everything as claimed in claim 10. Additionally, at col. 6, lines 20-40, Johnson teaches the semantic representation of the users input is sent to the applications code generator to allow the application to act on the code and retrieve the requested data, which reads on "the command is presented in a formal language such that a plurality of human utterances represent an action to be taken."

Art Unit: 2654

Regarding claim 17, Johnson, Eberman and Haddock teaches everything as claimed in claim 11. Additionally, at col. 3, lines 60-62, Johnson teaches maintaining item in focus information, which reads on “maintaining a current dialog focus.” At col. 5, line 61 continuing to col. 6, line 9, Johnson teaches concept/application table which reads on “a list of expected responses to provide a reference for determining the current context.”

Regarding claim 18, Johnson, Eberman and Haddock teach everything as claimed in claim 11. Additionally, at col. 4, lines 9-17 and col. 5, lines 9-11, Johnson teaches the natural language processor interacts with the user to clarify missing information or ambiguities, which reads on “querying a user for one of information needed to resolve the current context and information needed to take an appropriate action.

Regarding claim 19, Johnson discloses a multi-modal natural language interface for cross-application tasks which interprets user requests combining natural language input from the user with information selected from a current application and sends the request in the proper form to an appropriate auxiliary application for processing. Additionally, at col. 3, lines 60-62, Johnson teaches maintaining item in focus information, which reads on “maintaining a current dialog focus.” At col. 5, line 61 continuing to col. 6, line 9, Johnson teaches concept/application table which reads on “a list of expected responses to provide a reference for determining the current context.”

Johnson fails to specifically disclose, “The application associated with the command being one of a plurality of active applications.” However, implementation of a computer system, which allows a user to interact with a plurality of applications using natural language, was well known in the art.

Art Unit: 2654

In a similar field of endeavor, Eberman teaches an application user interface which allows the user to control the activities of multiple applications which are executed by the processors under control of the operating software system (col. 3, lines 25-39).

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to allow for the presented command to be associated with one of a plurality of active applications, as suggested by Eberman, for the purpose of allowing the user the convenience of accessing the processing of all applications to have a request processed automatically, without the user having to manually access any of the applications.

Johnson fails to specifically disclose "the dialog manager determining current context of the command by reviewing a multi-modal history of events." However, determining current context of user input responses by reviewing a history of events was well known in the art.

In a similar field of endeavor, Haddock discloses a multi-modal user interface for removing a referential ambiguity from a natural language input to a computer system. Specifically, Haddock teaches a user providing both a textual and a referential input and implements a history space or list of successive queries (col. 4, lines 61-66). The system functions to remove any ambiguity from a user query by referring to the history of the dialogue between the user and the system (col. 5, line 33 continuing to col. 6, line 64) to generate an output query that can be translated into a formal database query such as an SQL query which is then directed to the database to obtain a response. Haddock teaches the system is advantageous in providing a user-friendly way to resolve ambiguity in a natural language system (col. 2, lines 46-50).

Art Unit: 2654

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to implement determining current context of user input responses by reviewing a history of events as suggested by Haddock, for the purpose of providing a user-friendly way to resolve ambiguity in a natural language system, as also suggested by Haddock.

Additionally, at col. 4, line 17-53 and col. 5, line 23-29, Johnson teaches “changing dialog focus,” because the system ascertains that a different application should process the user request, and utilizes the different application’s resources to process the request, since the procedures to access a new or different application requires the system to access or focus on the new or different application. The system of Johnson performs this processing without requiring the user to perform the task of accessing the application, but the application is indeed accessed and opened to allow for the processing of the user’s request.

Additionally, at col. 5, line 55 continuing to col. 6, line 19, Johnson teaches referencing a concept/application table in which for a given input the application manager finds all concepts in the representation and determines each application registered with every concept in the representation to create an application set. The semantic representation is sent to the applications code generator to allow the application to act on the code and retrieve the requested data, which reads on “a control component adapted to select at least one method responsive to the commands received such that the system applies methods responsive to the commands for an appropriate application.”

Art Unit: 2654

Regarding claim 20, Johnson, Eberman and Haddock teach everything as claimed in claim 19. Additionally, at col. 4, lines 42-44, Johnson teaches the appropriate application is an auxiliary application that is opened in the background.

Regarding claim 21, Johnson, Eberman and Haddock teach everything as claimed in claim 19. Additionally, at col. 3, lines 37-42, Johnson discloses the input means can be via speech, typed or handwritten and includes point-and-click, touch and keyboard inputs, which reads on “multi-modal device is a computer.”

Regarding claim 22, Johnson, Eberman and Haddock teach everything as claimed in claim 19. Johnson does not specifically teach providing a linked list of all events in the multi-modal history. However, providing a linked history of dialog between a user and a system was well known in the art.

Haddock discloses the history of the dialog between the user and the computer system as a history space or list of successive queries (col. 4, lines 61-66), which reads on “linked list of all events.”

Therefore, it would have been obvious to one of ordinary skill at the time of the invention to modify the system of Johnson to implement a linked list of events in the multi-modal history, as suggested by Haddock, for the purpose of removing the ambiguity of a users input query, as also suggested by Haddock.

Regarding claim 23, Johnson, Eberman and Haddock teach everything as claimed in claim 22. Johnson does not specifically teach events in a multi-modal history include events linked by time, type, transaction, class or dialog focus. However, providing linked history events of a dialog between a user and a natural language system was well known.



Art Unit: 2654

Haddock discloses the history of the dialog between the user and the computer system as a history space or list of successive queries (col. 4, lines 61-66), which reads on "events linked by time."

Regarding claim 24, Johnson, Eberman and Haddock teach everything as claimed in claim 19. Additionally, at col. 5, line 55 continuing to col. 6, line 19, Johnson teaches referencing a concept/application table in which for a given input the application manager finds all concepts in the representation and determines each application registered with every concept in the representation to create an application set. The semantic representation is sent to the applications code generator to allow the application to act on the code and retrieve the requested data, which reads on "referencing all active applications using a component control to determine the at least one method which is appropriate."

Regarding claim 25, Johnson, Eberman and Haddock teach everything as claimed in claim 24. Additionally, at col. 6, lines 20-40, Johnson teaches the semantic representation of the users input is sent to the applications code generator to allow the application to act on the code and retrieve the requested data, which reads on "the command is presented in a formal language such that a plurality of human utterances represent an action to be taken."

### ***Response to Arguments***

2. Applicant's arguments with respect to claims 1-25 have been considered but are moot in view of the new ground(s) of rejection.

Art Unit: 2654

*Conclusion*

3. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than **SIX MONTHS** from the date of this final action.

Art Unit: 2654

4. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Angela A. Armstrong whose telephone number is 703-308-6258. The examiner can normally be reached on Monday-Thursday 7:30-5:00 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Marsha Banks-Harold can be reached on (703) 305-4379. The fax phone numbers for the organization where this application or proceeding is assigned are 703-872-9314 for regular communications and 703-872-9314 for After Final communications.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is 703-306-0377.

Angela A. Armstrong  
Examiner  
Art Unit 2654

AAA  
May 19, 2003

*Marsha D. Barb Harold*  
*SPE 2654*